Community Detection Methods in Social Network Analysis

Article *in* Journal of Computational and Theoretical Nanoscience · January 2014

DOI: 10.1166/asl.2014.5301

2 authors:

Andry Alamsyah
Telkom University
168 PUBLICATIONS **1,058** CITATIONS
SEE PROFILE

Budi Rahardjo
Bandung Institute of Technology
58 PUBLICATIONS **293** CITATIONS
SEE PROFILE

Advanced Science Letters
Vol. 4, 400–407, 2011

Community Detection Methods in Social Network Analysis

Andry Alamsyah[1,2], Budi Rahardjo[2], Kuspriyanto[2]
[1]School of Management Telecommunication and Media, Institut Manajemen Telkom
[2]School of Electrical Engineering and Informatics, Institut Teknologi Bandung
Bandung, Indonesia

In Social Network Analysis (SNA), community structure is an important feature of complex network. There are many researches on detecting community or cluster in graph with the objective to understand functional properties and community structures. Community detection early researches require global knowledge of network, which is not realistic to most real world network. Due to the increase of online social network, the new challenges are to develop methods to support community detection based on local information-only and network modularity. This paper present state of the art of methods in community detection research and propose the direction of future community detection research.

Keywords: Community Detection, Social Network Analysis, Complex Networks

1. INTRODUCTION

Social Network Analysis (SNA) represents actor as a node and relation as an edge in graph representation. The real world network, such as online social network exhibit the features of power law distribution, in which the network dominated by sparse connection. Dense connection inside the network can be viewed as community. A community in social network fulfills the criteria that there exist densely connected group of nodes, with only sparser connections between groups, as we can see in illustration in Figure 1. Analysis focus on whole network and ignore community structure may miss many interesting feature. Many paper in community detection research use different term for different context to describe community, such as *groups, subgroups, sub-network, clusters, cohesive groups* and *modules*. Detecting community remains a core problem in SNA. Finding out the groups inside a network also helps for other related social computing tasks.

Identifying communities in graph technically is finding node cluster in graph. Clustering in a network can be viewed as the strength-tie inside the community. We could say that research in community detection is to formalize the strong social groups based on social network properties.

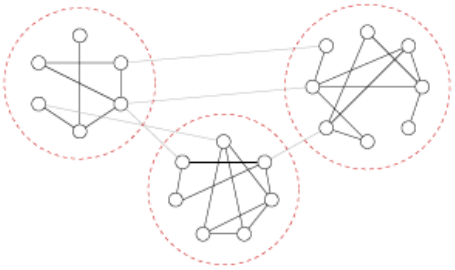

Fig. 1. Illustration of community structure inside a network taken from Newman [4]

Definition of community detection is subjective. Each method has different idea how to approach the problem of detecting community. Thus the result could be different for the same network.

The researches on community detection are extremely varied. They are based on a range different idea. The problems in community detection

*Email Address: andry.alamsyah@imtelkom.ac.id

1936-6612/2011/4/400/008
doi:10.1166/asl.2011.1261

research are finding the best approach from which have closest properties to real world network and the best methods that can handle scalability issue. This paper present issues, particularly, comparison between the basic community detection ideas and more advanced research based on real-world network. The issues that we discuss in this paper are local vs. global information, modularity network and overlapping communities. One issue that we have not discussed here is scalability or computational complexity. This issue will be presented on our next research based on comparative analysis community detection algorithm. However, some researchers has already extensively mapping state-of-the-art on scalability issue, one of them is Danon et. al [9].

2. COMMUNITY DETECTION METHODS

Classic approaches of finding communities in network borrow the idea of *graph partitioning* and *hierarchical clustering*. Graph partitioning approaches needs to know information about the global structure of network and determine in advance the number and size of subgroup that they want to get. Hierarchical partitioning is cluster analysis method in which the network of interest divided into several subgroups. The division is somewhat natural because it depends on node relationship inside the network than node properties itself. Node relationship is measured by similarity metric, such as *vertex similarity* [1] and *edge betweeness* [10]. Both metrics are using corresponding matrix, thus it has the drawbacks on computation complexity, when it come to large-scale network.

Community detection methods and algorithms are discussed in [1][2]. From many different ideas and perspective, the community detection based research roughly categorized into four approaches. 1. *Node-Centric* 2. *Group-Centric* 3. *Network-Centric* 4. *Hierarchical-Centric*.

Node-Centric criteria require each node in a group to satisfy properties such as *complete mutuality* and *reachability*. *Clique*, a fully connected subgroup, indicates complete mutuality. In many common situations, clique is hard to find, because the definition is very strict. Steps in complete mutuality including finding clique of size *k*, and then prune those nodes with *k-1* degree. Complete mutuality is a good measure of tie-strength inside the subgroup, but it is a NP-Hard problem, therefore it is practically have little use. Reachability among nodes happened if there exist path between those nodes. Large component can be easily formed in a social network, while others are minor communities or even singletons. More effort is needed to find communities inside large component. The most useful metrics for reachability are *k-clique* and *k-club*. K-Clique is maximal subgroups in which the largest geodesic distance between any of two nodes is no greater than k. K-Club restricts the geodesic distance within the group to be no greater than k. It is often requires combinatorial optimization and it remains a challenge to generalize them in large-scale network. In Figure 2 we illustrate: clique is {A, B, C}, 2-cliques are {A, B, C, D, E} and {B, C, D, E, F}, 2-clubs are {A, B, C, D}, {A, B, C, E}, and {B, C, D, E, F}.

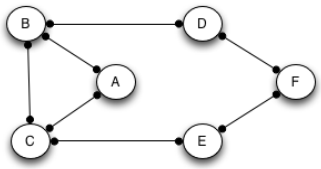

Fig. 2. Illustration about clique, k-cliques and k-clubs

Group-Centric considers the connection within the group as a whole. The group is required to satisfy *density-based* group requirement, while some nodes inside the group may have low connectivity. A measure for density-based group is *quasi-clique* γ [12]. A subgroup $G_s(V_s, E_s)$ is γ-dense if

$$\frac{E_s}{V_s(V_s-1)/2} \geq \gamma$$

Group-centric approach does not guarantee reachability for each node. It allows the degree of node vary, hence it is suitable for complex networks and large-scale networks. The objective of density-based group is to found the maximal quasi-clique easily. The steps for discovering communities applied as follows: 1. Search randomly a maximal quasi-clique in a sub-network, Apply greedy approach to expand quasi-clique by encompassing those high-degree neighboring nodes until the density drop below γ. 2. Prune nodes and edges that have degree less than kγ, because it is unlikely contribute to larger quasi-clique by including such a node. This process is iterated until network reduced in a reasonable size so that a maximal quasi-clique can be found directly.

Network-Centric objective is to create numbers of disjoint sets from the network. Using several criterions, network-centric considers the connection of nodes globally. There are 5 known methods for this approach. They are *node similarity, latent space model, block model approximation, cut minimization / spectral clustering, modularity maximization*.

Node similarity is defined by how similar their interactions are. Two nodes are *structurally equivalent* if they connect to the same set of nodes. This measure is too restrictive and rarely occurs in large-scale network. Alternative relaxed approaches for two nodes v_i and v_j are *jaccard similarity* $J(v_i, v_j)$ and *cosine similarity* $C(v_i, v_j)$, which can be written as:

$$J(v_i, v_j) = \frac{|N_i \cap N_j|}{|N_i \cup N_j|} \text{ and } C(v_i, v_j) = \frac{|N_i \cap N_j|}{\sqrt{|N_i||N_j|}}$$

Where N_i and N_j denotes the set of nodes of direct neighbors of node v_i and v_j. Both similarities have value between 0 and 1. Once similarity is determined, we apply *k-means clustering* [2] to every node in the network, until all nodes joined to the closest groups/centroid. The process of computing nodes similarity totaling $O(n^2)$. It is time consuming when network is very large.

Latent space models ideas is to transform nodes in the network into low-dimensional Euclidian space such that similarity and distance are kept in the new space. Once the transformation done, we begin clustering network in the low-dimensional space using methods like *k-means*. The transformation process is using *multi dimensional scaling* (MDS). Typical processed in MDS are: 1. Construct proximity matrix between each node in the higher *n*-dimension $D \in R^{nxn}$. 2. Find $S \in R^{nxl}$ in lower *l*-dimension using formula from [x]:

$$SS^T \approx -\frac{1}{2}\left(I - \frac{1}{n}\,11^T\right) D \left(I - \frac{1}{n}\,11^T\right) = \Delta D$$

The objective of MDS is to minimize $\|SS^T - \Delta D\|$. Suppose V contains the top l eigenvectors ΔD with largest eigenvalues, Λ is diagonal matrix of top l eigenvalues $\Lambda = \text{diag}\,(\lambda_1, \lambda_2, \ldots, \lambda_l)$. The optimal S is $S = V\Lambda^{1/2}$. The cluster number is defined by number of top eigenvector.

Block model approximation approximates a given network by a block structure. The steps are including: 1. Create a block structure from an interaction matrix (adjacency matrix) A. 2. The block structure contains S, a block indicator matrix which corresponds to top k eigenvectors of A. 3. Apply k-means clustering to S to discover the community partition. The key objective of this method is to minimize the difference between an interaction matrix (adjacency matrix) and a block structure or we can write as $\min\|A - S\Sigma S^T\|$, where S is community indicator matrix that we set in advanced. Each block represents one community. In Figure 3, we see the illustration of a given network with an adjacency matrix A and its block structure $S\Sigma S^T$.

-	1	1	1	0	0	0	0	0
1	-	1	0	0	0	0	0	0
1	1	-	1	0	0	0	0	0
1	0	1	-	1	1	0	0	0
0	0	0	1	-	1	1	1	0
0	0	0	1	1	-	1	1	0
0	0	0	0	1	1	-	1	1
0	0	0	0	1	1	1	-	0
0	0	0	0	0	0	1	0	-

(a)

1	1	1	1	0	0	0	0	0
1	1	1	1	0	0	0	0	0
1	1	1	1	0	0	0	0	0
1	1	1	1	0	0	0	0	0
0	0	0	0	1	1	1	1	1
0	0	0	0	1	1	1	1	1
0	0	0	0	1	1	1	1	1
0	0	0	0	1	1	1	1	1
0	0	0	0	1	1	1	1	1

(b)

Fig. 3. (a) Sample of adjacency matrix A and (b) the block structures of matrix A

Cut minimization is derived from the problem in graph partition. Cut is the total number of edges between two disjoint sets of nodes. Graph partition objective is to discover partition such that the cut is minimized. Two common variant used are *ratio cut* and *normalized cut.* Ratio cut represents number of nodes in a community. Normalized cut represents number of interactions inside group. Let $\pi = (C_1, C_2, \ldots, C_k)$ be a graph partition such that $C_i \cap C_j = \emptyset$ and $\cup_{i=1}^{k} C_i = V$. The ratio cut $R(\pi)$ and normalized cut $N(\pi)$are defined as:

$$R(\pi) = \frac{1}{k}\sum_{i=1}^{k}\frac{cut(C_i, \bar{C_i})}{|C_i|}$$

$$N(\pi) = \frac{1}{k}\sum_{i=1}^{k}\frac{cut(C_i, \bar{C_i})}{vol\,(C_i)}$$

Where $\bar{C_i}$ is the complement of C_i, and $vol\,(C_i) = \sum_{v \epsilon C_i} d_v$. Both objectives attempt to minimize the number of edges between communities, yet avoid the bias of trivial-size communities like singletons. Cut minimization can be relaxed into minimum trace problem using *Graph Laplacian.* The same from previous network-centric method, we apply k-means clustering algorithm. The last process we call as Spectral Clustering.

Modularity maximization measures the group interactions compared with the expected random connections in the group. In a network with m edges, for two nodes with degree d_i and d_j, expected random connections between them are $d_i d_j/m$. The interaction utility in a group is $\sum_{i \epsilon C, j \epsilon C} A_{ij} - d_i d_j/m$. This utility also measures how far the true network interactions between nodes i and j deviates from expected random connections. At last to partition network into several groups, we maximize *modularity*, which is defined as:

$$Q = \frac{1}{2m}\sum_{l=1}^{k}\sum_{i \epsilon C_l, j \epsilon C_l}(A_{ij} - d_i d_j/2m)$$

1936-6612/2011/4/400/008
doi:10.1166/asl.2011.1261

The coefficient $1/2m$ normalize modularity value between -1 and 1. Modularity = 0, if all nodes are all clustered into one groups. It can automatically determine the optimal number of clusters. At the final steps, we find the k-top eigenvectors of the modularity matrix *B,* which is built based on:

$$Q = \frac{1}{2m}\sum_{l=1}^{k} s_l B s_l$$

Hierarchy-Centric objective is to build a hierarchical structure of communities based on network topology. The methods facilitate detection of communities at different level from top-bottom and bottom-up approach. There are two types of hierarchical clustering: *divisive* and *agglomerative*. The steps in divisive hierarchical clustering are 1. Partition the nodes into several smaller sets. 2. Each set is further partitioned into smaller sets. One particular metric to use is *edge-betweenness*, which defined as the number of shortest paths that pass along one edge. At each iteration, it recursively remove the edges that have low edge-betweenness or the weakest tie. Agglomerative hierarchical clustering is the opposite of divisive methods. They initiate each node as community, and then choose two communities satisfying certain criteria such as modularity or node similarity; in the end we merge both communities. This process is iterated until there are no more nodes to merge. Agglomerative can be very sensitive to the node processing order and merging criteria adopted. Divisive clustering are more stable but computationally expensive.

The methods presented in this chapter are the base to intensify community detection research. Some of the methods will be developed to accommodate properties of real-world network, which some examples will be discussed in next chapter.

3. COMMUNITY DETECTION WITH LOCAL INFORMATION-ONLY, MODULARITY AND OVERLAPPING COMMUNITIES

The global knowledge of the network structure sometimes is impossible to find. In the case of online social network, they contain millions nodes and edges. For example, users in *Youtube* can be categorized based on the number of video they post, comment they make, their friendship, favorites list and some others categories. In short, in one social network we can cluster interactions into several groups based on type of interactions we want. This description adds up the complexity of having global knowledge about the network.

Chen et al. [3] propose methods based on *Iterative Local Expansion.* This method only need local information about the node, hence it is particularly useful for large-scale networks. The process called as one-node-at-one step discovery process to find node $s_i \epsilon S$, where cluster *S* adjacent to cluster *D*. In order to get global information of network *G*, the process visit some neighbor node s_i of *D* (where $s_i \epsilon S$) and obtain a list of adjacency of s_i. As a result of, s_i is removed from *S* and becomes a member of *D*. This process similar to web crawling system explores the WWW. Local modularity is presented in this paper to make sharp boundary between the communities. Another modularity, which contain ratio of number internal edges and number external edges, is proposed for local community evaluation.

De Meo et al. [5] propose methods exploits novel measure of *k-path* edge centrality [6]. This technique allows to efficiently computing edge ranking in large network. The discovery of community structure is adopting strategy inspired by well-known state-of-the-art *Louvain Methods* [6][13]. The k-path edge centrality $L^k(e)$ of edge *e* in graph *G = (V, E)* defined as the sum, over all possible sources nodes *s*, percentage of times that a message originated from *s* traverse *e*. Louvain methods strategy based on local information. It is based on two steps: 1. Each node is assigned to a community chosen in order to maximize the network modularity. 2. Makes new network consisting of nodes that are those communities previously found. Then the process iterates until a significant improvement of the network modularity found.

The number of research in overlapping communities is not as many as community detection. One of which is important is based on local oriented efficient detection [7]. This method significantly superior than the previous approach for detecting overlapping community detection using *Clique Percolation Method* (CPM) [8]. CPM algorithm detects communities by searching for adjacent *k-cliques*. CPM is suitable for network with dense connected parts but fails to terminate in many large social networks. An algorithm for overlapping community discovery known as *local fitness metric* (LFM) is based on local optimization of fitness function. The drawback of LFM is occurrence of the loop due to dysfunction of the fitness metric as well as random seed selection used. *Local oriented fitness optimization* (LOFO) is introduced to improve the detecting quality and computation efficiency of LFM. LOFO local oriented scheme based on clustering coefficient and several efficiency-enhancing schemes. The experiment result shows LOFO significantly outperforms LFM and CPM. State of the art of current development of overlapping communities research is discussed in [9], including comparison several algorithm and evaluation metric.

4. CONCLUSION

The community detection research direction recently moves towards more realistic approach for real world network, which is complex and large-scale [1]. Node local information and its adjacency combine with

network modularity are often employed. The ability to detect nodes / group in overlapping communities is also important, since in real-world network, nodes membership is not limited to only one communities. One important aspect that we have not discussed here is the scalability or computational complexity, for the reason we mention in the introduction section. In the mean time, we can see in Danon et al. [9] paper that give a comparative complexity analysis several community detection algorithm. The four issues above (*local, modularity, overlapping and scalability*) will be the main direction of the future community detection research. However the nature of community detection, which sometimes qualitative and subjective has posed certain problems, one of them is behavior based on the network size [14].

There are also many others line of researches of community detection, some are still in early stages, some are less popular, but they are also promising in the future. Those methods [1] are based on random walks, spin models, statistical inference, label propagation and also power graph analysis [15].

REFERENCES

[1] S. Fortunato. Community Detection in Graph. *Physics Report 486* (2010) 75-174

[2] L. Tang, H. Liu. Graph Mining Applications to Social Network Analysis. In C.Aggarwal and H. Wang, editor, *Managing and Mining Graph Data, chapter 16. Springer* (2010) 487-513

[3] J. Chen, O.R. Zaiane, R. Goebel. Detecting Communities in Large Networks by Iterative Local Expansion. Proceeding in *2009 International Conference on Computational Aspects of Social Networks* (2009) 105-112

[4] M.E.J Newman, M. Girvan. Finding and Evaluating Community Structure in Network. *Phys. Rev.* E. 69, 026133 (2004)

[5] P. De Meo, E. Ferrara, G. Fiumara, A. Provetti. Generalized Louvain Method for Community Detection in Large Scale. *11th International Conference on Intelligent System Design and Application* (2011) 88-93

[6] E. Ferrara, P. De Meo, G. Fiumara. A Novel Measure of Edge Centrality in Social Network. *Journal Knowledge-Based System, Vol. 30* (2012) 136-150

[7] S. Liang, Y. Guo. Local Oriented Efficient Detection of Overlapping Communities in Large Network. *International Conference on Cyber-Enabled Distributed Computing and Knowledge Discover* (2012) 31-38

[8] G. Palla, I. Der'enyi, I. Farkas, t. Viscek. Uncovering The Overlappg Community Structure of Complex Networks in Nature and Society. *Nature-435* (2005) 814-818

[9] L. Danon, A. Diaz-Guilera, J. Duch, A. Arenas. Comparing Community Structure Identification. *Journal of Statistical Mechanics*, *P09008* (2005)

[10] M.E.J. Newman. Fast Algorithm for Detecting Community Structures in Networks. *Phys. Rev. E. 69, 066133* (2004)

[11] L. Tang, H. Liu. Community Detection and Mining in Social Media, *Morgan & Claypool* (2010) 31-45

[12] J. Abello, M.G.C. Rasende, S. Sudarsky. Massive Quasi-Clique Detection. *Proceeding LATIN'02 Proceedings of the 5th Latin American Symposium on Theoritical Informatics, Springer* (2002) 598-612

[13] V.D. Blondel, J.L. Guillaume, R. Lambiotte, E. Lefebvre. Fast Unfolding of Communities in Large Networks. *Journal of Statistical Mechanics: Theory and Experiment*, *P10008* (2008)

[14] J. Leskovec, K.J. Lang, M. Mahoney. Empirical Compariosn of Algorithm for Network Community Detection. *Proceeding WWW'10 Proceedings of the 19th International Conference on World Wide Web* (2010) 631-640

[15] G. Tsatsaronis, M. Reimann, I. Varlamis, O. Gkorgkas, K. Norvag. Efficient Community Detection using Power Graph Analysis. *Procedding LSDS-IR'11 Proceddings of the 9th Workshop on Large-Scale and Distributed Informational Retrieval* (2011) 21-26

1936-6612/2011/4/400/008
doi:10.1166/asl.2011.1261